ERB-969

NRCC No. 23374

National Research Council Canada
Conseil national de recherches Canada

Division of Electrical Engineering
Division de génie électrique

ON THE CHORD PROPERTY AND ITS APPLICATIONS

S.H.Y. HUNG

MAY 1984

Canada

85-3235/

Copyright 1984 by
National Research Council of Canada

Permission is granted to quote short excerpts and to reproduce figures and tables from this report, provided that the source of such material is fully acknowledged.

Copyright 1984 par
Conseil national de recherches du Canada

Il est permis de citer de courts extraits et de reproduire des figures ou tableaux du présent rapport, à condition d'en identifier clairement la source.

Additional copies are available free of charge from:

Publications Office, Room 301
Division of Electrical Engineering
National Research Council of Canada
Ottawa, Ontario, Canada
K1A 0R8

Des exemplaires supplémentaires gratuits peuvent être obtenus à l'adresse suivante:

Bureau des publications, Pièce 301
Division de génie électrique
Conseil national de recherches du Canada
Ottawa (Ontario) Canada
K1A 0R8

ON THE CHORD PROPERTY AND ITS APPLICATIONS

S.H.Y. Hung

ERB-969
NRCC No. 23374

MAY 1984

TABLE OF CONTENTS

		Page
ABSTRACT/RÉSUMÉ		v
I.	INTRODUCTION	1
II.	SOME NECESSARY CONDITIONS FOR THE CHORD PROPERTY	2
III.	CURVATURE CODE AND ITS APPLICATIONS	3
IV.	ANGLE-FINDING ALGORITHM AND C-SEGMENTS	9
V.	THE RUN LENGTH REPRESENTATION AND THE CS PRINCIPLE	11
VI.	NECESSARY AND SUFFICIENT CONDITION FOR THE CHORD PROPERTY	15
VII.	REMARKS AND SOME IMPORTANT LEMMAS	17
VIII.	THE PROOF OF THE MAIN THEOREM T5	23
IX.	POSSIBLE CRITICAL POINTS IN A C-SEGMENT	28
X.	SOME PRACTICAL CONSIDERATIONS	32
XI.	CONCLUSION	35
REFERENCES		35

ABSTRACT

A necessary and sufficient condition for a digital arc to have the chord property, as well as a simple algorithm for checking this condition, is given. It offers a much simpler alternative to examine the straightness of a digital line segment rather than checking the chord property directly. The final result is a set of critical points which divide a digital arc into straight segments such that each of them is guaranteed to have the chord property. If the only segment after the segmentation is the arc itself, then the digital arc is straight. The necessary and sufficient conditions for the chord property are also the most comprehensive and precise expression about the third criterion of digital straight line given by H. Freeman.

RÉSUMÉ

On présente un condition nécessaire et suffisante pour qu'un arc numérique soit considéré comme une corde, de même qu'un algorithme simple pour vérifier ce condition. De cette façon, il est beaucoup plus simple de vérifier la rectitude d'un segment de ligne numérique; il n'est pas nécessaire de vérifier directement les propriétés d'une corde. Le résultat final est un ensemble de points critiques divisant l'arc numérique en segments rectilignes qui présentent tous les propriétés d'une corde. Lorsque le seul segment résultant de la division en segments est l'arc lui-même, on peut conclure que l'arc numérique est rectiligne. Les conditions nécessaires et suffisantes pour qu'un segment soit une corde constituent aussi l'expression la plus complète et précise du troisième critère de ligne droite numérique donné par H. Freeman.

ON THE CHORD PROPERTY AND ITS APPLICATIONS

S.H.Y. Hung

I. INTRODUCTION

In pattern recognition and scene analysis, there are many occasions when we want to determine whether a digital arc is straight or not. For many applications, such as the production of cartographic data bases, waveform analysis, and perimeter estimation, it is preferable to access the curve as a set of connected straight lines or a polygonal line.

There are many techniques for segmenting a digital arc into straight segments to satisfy different criteria for different applications (1-11). However, most of them cannot avoid the time-consuming distance and angle calculations, or other computions. Many of them are good for certain applications but not for others. Simpler schemes that can be good for all occasions are always desirable.

Rosenfeld (12) gave the most concise and vigorous criterion, the chord property, for determining the straightness of a digital arc. However, the chord property has not been popularly used for this purpose. That is due to the large amount of computation that is required to test it. As well, it is often too restrictive for practical applications. Many digital arcs obtained in the real environment do not have this ideal property.

In Hung and Kasvand (13) a simple scheme has been used to segment a digital arc into straight segments, trying to satisfy human visual criteria. This method can be refined to segment a digital arc such that each segment guarantees to have the chord property. Since this method is based on the first and second order of curvature codings which are derived from the chain codes of a digital arc, only simple arithmetic is required and the computing time is almost negligible. Once we divide a digital arc into such connected straight segments, we have to deal with only those segments rather than points. Whether or not this is too restrictive can be dealt with in subsequent steps.

Basically, this scheme can be divided into two parts. The first part is essentially an angle-finding algorithm that finds all obvious angles and bendings, which will be singled out under almost any circumstance. The segments divided by those points will be quite smooth and have only very gentle curvatures. In some applications that is all that is needed and the process can stop here. In the second part, we examine those smooth segments to determine whether, in order to have the chord property, any

further dividing is needed. The final result is a set of connected straight segments which have the chord property. If the only segment after the segmentation is the arc itself, then the digital arc is straight.

II. SOME NECESSARY CONDITIONS FOR THE CHORD PROPERTY

The digitization scheme supposedly used in this report is essentially the standard grid-intersect method of digitizing curves (14). Given a coordinate grid superimposed on the curve, then whenever the curve crosses a grid line, the grid point nearest to the crossing becomes a point of the digitization of the curve. When the crossing is exactly midway between two grid points, we always round down to the one having the smaller value.

A point whose coordinates are integers is called a DIGITAL POINT. A DIGITAL ARC S is a connected set of digital points all but two of which have exactly two 8-neighbours in S, while two exceptions (the endpoints) each have exactly one 8-neighbour in S (12). A digital arc is STRAIGHT if it is the digitization of a straight line segment.

The definition of the CHORD PROPERTY given in ref. 12 can be stated as follows:

> A digital arc S is said to have the chord property if for each point $p = (x, y)$ on $\overline{cd}$ where c and d are any two points in S, there is a point $e = (h, k)$ of S such that $\max\{|x - h|, |y - k|\} < 1$, where $\overline{cd}$ is the line segment between c and d. The points c, d, and e are all digital points.

Rosenfeld in ref. 12 showed that a digital arc is straight iff it has the chord property; and if a digital arc S has the chord property, it should have the following important properties:

C1. All runs in S can have at most two directions differing by 45°.

C2. No two consecutive runs can both have length >2 (a special case of C3).

C3. S cannot have runs of length >2 in two different directions, even if they are not consecutive, i.e., in one of the directions, all runs must have unit length.

C4. In the other direction, the runs in S can have only two lengths which are two consecutive integers.

C5. One of the run lengths can occur only once.

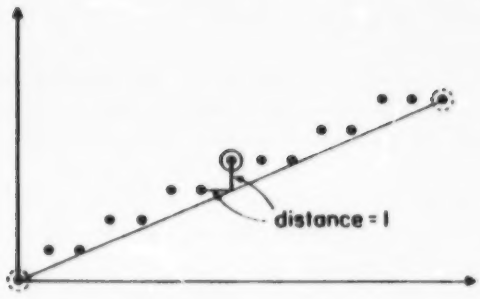


Figure 1

However, conditions C1-C5 are necessary but not sufficient. They consider almost everything except that the distribution of the other run length can occur more than once. The digital arc in Fig. 1 is an example which satisfies all C1-C5 but does not have the chord property.

Although C1-C5 are not sufficient conditions, they do provide a much easier alternative, at least partially, for determining the straightness of a digital arc rather than checking the chord property directly.

The condition C1-C3 consider the obvious bendings which will be seen as breaking points under almost any standard. Once a segment satisfies C1-C3, it can only have two directions differing by 45° (if it has only one direction, it is always straight and no further consideration is needed). In one of those directions, all runs must be of unit length. If there are any more bendings that will be significant enough to violate the chord property, then they must be caused by the run lengths and their distributions in the other direction. All these will be considered by C4, C5, and whatever conditions that we must add to them to make them become sufficient conditions for a digital arc to have the chord property. Those further considerations have to apply to only the digital segments which already satisfied the C1-C3; they are quite smooth and will be easy to deal with. This is one of the reasons that the procedure is simple and effective.

In the next section, we will describe a scheme that is a refinement of the scheme in ref. 13 and will be used to check C1-C3.

III. CURVATURE CODE AND ITS APPLICATION

We assume that every digital arc we are dealing with is chain coded (14). However, we prefer to use the numbers 1, 2, 3, ..., 8 rather than 0, 1, 2, ..., 7 as indicated in Fig. 2. The 0 is used to indicate the terminating point of a digital arc. A digital arc S of length n can be written as

$$S = [D(1), D(2), D(3), \ldots, D(n)]$$

where D(i), i = 1, ..., n are the chain codes of digital points D(i), i = 1, ..., n (we are using the same symbol for a digital point and its chain code). D(n) = 0 is the only zero chain code in the whole arc.

Definition: The CURVATURE CODE C(i) of a point D(i) of a digital arc is a value that is determined by the relative positions of the consecutive points D(i - 1), D(i), and D(i + 1), as indicated in Fig. 3. The shadowed position at the tail of the arrow is the position of D(i - 1), and at the head of the arrow is D(i). Those numbers in the right, top-and-right are the curvature codes for D(i), if D(i + 1) is at the corresponding position.

From the definition of a digital arc, it is obvious that those are all the possible values for the curvature codes. The curvature code is a quantization of curvature. It indicates the change of direction of a digital arc at the point.

Definition: We call a digital point with zero curvature code a ZERO POINT; otherwise, it is a NON-ZERO POINT.

The end points of a digital arc are always seen as non-zero points; even their curvature codes are zeros. We also call the curvature code of a point its VALUE. Throughout this paper, it will be the same if we say that a point has a non-zero curvature code or a point has a non-zero value.

The curvature codes can be derived directly from the chain codes. Let

$$S = [D(i), i = 1, \ldots, n]$$

be the arc and its chain codes, and C(i) be the curvature code of D(i) in S.

$$C(1) = C(n) = 0$$

For i = 2, 3, 4, ..., n - 1

$$C(i) = D(i) - D(i - 1) \qquad \text{if} \quad -2 < D(i) - D(i - 1) < 2$$

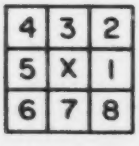


Figure 2

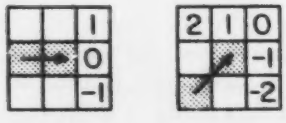


Figure 3

$$C(i) = D(i) - D(i-1) + 8 \quad \text{if} \quad D(i) - D(i-1) < -2$$

$$C(i) = D(i) - D(i-1) - 8 \quad \text{if} \quad D(i) - D(i-1) > 2$$

For example, see Fig. 4. The numbers in the squares are the chain codes, the numbers below are curvature codes as indicated by C(i).

By using the curvature code we can define a RUN in a digital arc as:

Definition: A segment of a digital arc which begins and ends at two nearest non-zero points (there is no non-zero point between) is called a run and denoted as R.

Definition: If there is no zero point between two non-zero points, then the length of this run is 1 (unity). If there are zero points between, then the length of this run is equal to the number of zero points plus 1. The LENGTH OF A RUN is denoted as L.

Definition: The VALUE OF A RUN is the sum of curvature codes of its two endpoints. The exceptions are the first and the last runs of the arc; their values are defined to be zero. The value of a run is denoted as V.

The value of a run is a measure of significance of the local curvature change. A zero value indicates a locally insignificant change in direction of the arc, except in the case of an isolated single non-zero point, which always indicates a significant change in direction.

Each non-zero point must be the endpoint of two consecutive runs. It is associated with the values of two runs, one in the front and one behind. We call them the FRONT VALUE and the REAR VALUE of this non-zero point.

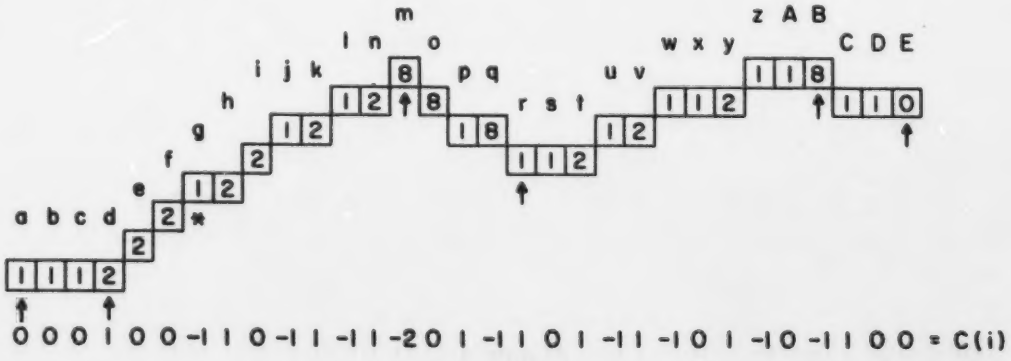


Figure 4

Definition: A CRITICAL POINT is a point in a digital arc at which a significant bending of the arc takes place, i.e., at such a point a bend appears such that one can no longer see the arc as straight, and a point is chosen to divide it into two segments. Otherwise, the chord property will be violated.

We will use the chord property as the criterion for finding all the critical points in a digital arc, such that every segment between two critical points will have the chord property. The first part of our method is essentially an angle-finding algorithm. In this part we will find all the obvious critical points by checking the conditions C1-C3. Any critical points that are caused by reasons other than C1-C3 will be considered later. We will prove some theorems first; the whole angle-finding algorithm is based on them.

Theorem T1: A point with a curvature code of 2 or -2 is always a critical point.

Proof: In order to have a curvature code 2 or -2, the incoming run to, and the outgoing run from this point must differ by 90°. The C1 has been violated, so these two runs must be separated by a critical point. The only point between these two runs is the point itself and it must be critical (see Fig. 5a).

Theorem T2: An isolated single non-zero point is always a critical point.

Proof: If this point has a value 2 or -2, then we are finished. It must be considered only if the point has a value 1 or -1. In order to have such an isolated single non-zero point, the

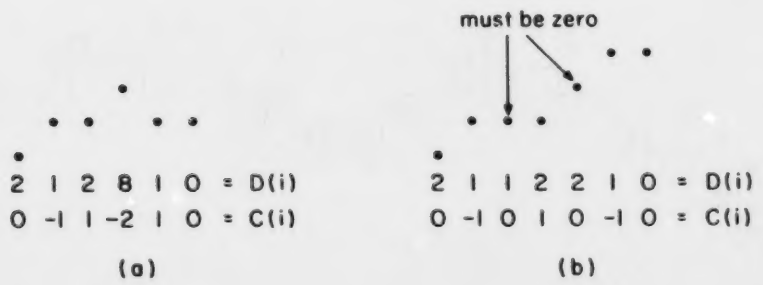


Figure 5

runs in both sides of this point must differ by 45° and both have length >2 (see Fig. 5b.). This violates C2.

This is why we keep C2 as a separate item, because it corresponds to an important case, namely, the isolated single non-zero point.

Definition: A NON-ZERO SEGMENT (NZS) is a string of consecutive non-zero points which is isolated by zero points from the rest of the arc. The length of such a segment is equal to the number of non-zero points in it. The sum of the curvature codes of all points in a NZS is called the GROUP SUM of this segment.

The group sum of a NZS indicates the change in direction between the incoming run and outgoing run of this segment.

A special kind of NZS which consists of alternative 1 and -1, and with the values of all its runs zero, is called a SEQUENCE OF PERFECTLY CANCELLING PAIRS (SPCP). A SPCP itself can always be seen as straight. If the length of a SPCP is an even number, then its group sum is 0 and we do not have to consider it at this point. If the length is an odd number, then its group sum will be 1 or -1. That means the incoming run and the outgoing run of this SPCP are both of length >2 and their directions differ by 45°. This violates C3, so one of the non-zero points in this sequence must be critical. We have:

Theorem T3: One of the non-zero points in a SPCP of odd length must be chosen as a critical point.

For example, see Fig. 6. Obviously, T2 is a special case of T3; an isolated single non-zero point is a SPCP of length 1.

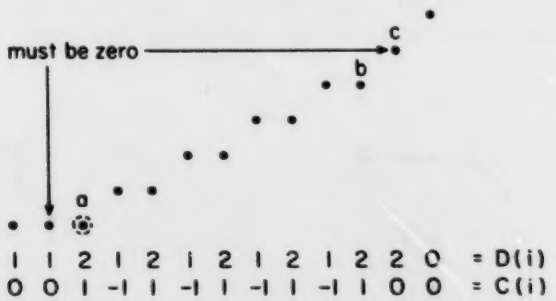


Figure 6

Theorem T4: If the value of a run is non-zero, then one of its points must be chosen as a critical point.

Proof: If one or both of the endpoints of this run are of value 2 or -2, then we are finished. Therefore, we assume that its endpoints have values 1 or -1. In order to have a non-zero sum, both endpoints must have the same value 1 or -1 (see Fig. 7a). In both cases, the incoming run to point a and outgoing run from point b differ by more than 45°. One of the points between, including a and b, must be critical.

Corollary 1: If the run with non-zero value is of length 1, then one of its endpoints has to be critical.

Proof: There is no other point in it.

Corollary 2: If there are two consecutive runs with non-zero values, the common endpoint is critical.

Proof: For both runs, one of the points must critical by T4. The common endpoint between them is the natural choice.

For example, see Fig. 7b. Consider a, b, c, d, and e first. If both V(b, c) and V(c, d) are non-zero, we need only to choose c to be critical. If V(e, f) is also non-zero, then to d both V(c, d) and V(d, e) are non-zero. But we do not have to take d to be critical, since c is already critical and it satisfies the requirement for V(c, d) to be non-zero. The runs R(c, d) and R(d, e) differ by only 45°. The problem is caused by R(e, f) and R(c, d) and only e needs to be critical. V(c, e) can be seen as an isolated one and always chooses the second endpoint to be critical. There is no need to have two critical points appearing immediately beside each other, unless one or both of them have curvature codes 2 or -2.

A simple angle-finding algorithm for checking the C1-C3, which is based on those theorems, will be described in the next section.

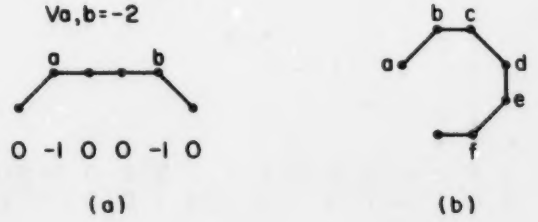


Figure 7

IV. ANGLE-FINDING ALGORITHM AND C-SEGMENTS

Firstly, we are going to calculate the curvature codes from the chain codes of a digital arc. Then we are going to consider all non-zero points by going through all NZS's one by one. Although a zero point can be chosen as a critical point, sometimes it may even reduce the number of critical points. In Fig. 8, point b or c can be chosen as a critical point, and only one of them must be critical. If we choose a or d, we must take both of them (this will be explained later). However, for convenience and to try to make the algorithm as simple as possible, we consider only the non-zero points. Most bendings occur at a non-zero point.

The endpoints of a digital arc are always critical, and the rest of non-zero points will be considered in the following steps:

1. If a NZS is of length 1, i.e., it is an isolated single non-zero point, then it must be critical.

2. Any pcint in a NZS which has length more than 1 can be checked one by one according to the following steps:

   2.1. If the current non-zero point has value 2 or -2, it must be critical. Otherwise, go to 2.2.

   2.2. All points coming to this step must be value of 1 or -1.

      2.2.1. Check the front value of this point. If it is zero, then process the next non-zero point. Otherwise, go to 2.2.2.

      2.2.2. Check the previous non-zero point. If it is a critical point already, then process the next one. Otherwise, this current point must be critical.

3. If, after step 2, none of the non-zero points has been chosen as critical, then we have a segment of alternative 1 and -1, i.e., a SPCP.

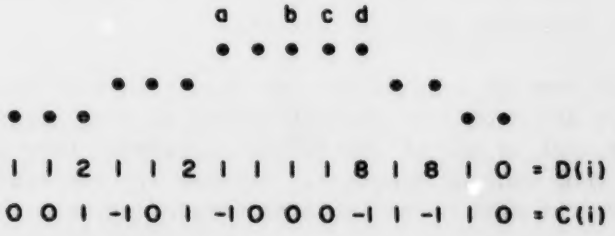


Figure 8

3.1. If the length of this SPCP is an even number, we can go to process the next NZS. Otherwise, go to 3.2.

3.2. The NZS which comes to this step must be a SPCP with odd length. In Fig. 6, we can see that the incoming run and the outgoing run of this SPCP differ by 45°. This SPCP can be attached to any one of them, but not both. It is better to attach it to the run in the diagonal direction, and we choose the point a to be critical. This can be done by testing the chain code of D(c). If D(c) is an odd number, then b should be critical. Otherwise, a is critical, i.e., the endpoint of this SPCP, which is next to a zero point with odd chain code, is critical.

The end result of steps 1, 2, and 3 is a set of critical points such that they divide the digital arc into segments. These segments with two consecutive critical points as their endpoints are called C-SEGMENTS. In Fig. 4, the points indicated by a small arrow are the critical points found by these steps. The segments between two such points are C-segments. The points a and E were chosen because they are the endpoints. Point d is an isolated single non-zero point. Point m has a value of -2. Point r is a non-zero point in a SPCP next to a zero point (s) with odd chain code, and the SPCP (pqr) has an odd length but no point can be found to be critical by steps 1 and 2. Point t is not critical; even the V(r, t) is not zero, because r is critical already. Point B is critical because V(z, B) is non-zero and z is not critical.

It is easy to see that each C-segment must satisfy C1-C3. If there are two runs in a C-segment which differ by more than 45°, then the sum of values of all non-zero points between these two runs must be greater than 1 or less than -1. If there is an isolated single non-zero point, or one of those non-zero points has value 2 or -2, then it must be chosen as a critical point by step 1 or step 2.1 and these two runs cannot be in the same C-segment. If there is no point with value 2 or -2, then the number of non-zero points with value 1 and the number of non-zero points with value -1 must differ by 2 or more. Therefore, some values of runs must be non-zero, and there must be at least one point being chosen by step 2.2. to be critical. So there cannot be two runs in a C-segment differing by more than 45°; thus a C-segment must satisfy C1.

If there are runs of lengths greater than 1 in both directions, we can assume that there are a pair of runs of length 2, with one of them in each direction. This must be one of the following cases: they are consecutive or separated by some runs of length 1. If they are consecutive runs, then their common endpoint must be an isolated single non-zero point, and be

chosen by step 1 to be a critical point. If they are not consecutive, then the NZS between must be a SPCP with odd length (otherwise, there must be some points chosen to be critical, or these two runs are in the same direction), and one of its point must be chosen as critical by step 3. Therefore, in one of the directions all runs can have unit length only, and a C-segment must satisfy C2 and C3.

In the following sections we will describe the algorithm that will be used to further examine the C-segments. To guarantee that the final segments will have the chord property, we will first give the necessary and sufficient conditions for a digital segment which already satisfies C1-C3 and has the chord property. Then a simple procedure will be given to check these conditions.

V. THE RUN LENGTH REPRESENTATION AND THE CS PRINCIPLE

Starting from here, we have only to deal with C-segments. A C-segment always satisfies C1-C3 and can be described as follows: it has only two directions differing by 45° (if it has only one direction, then it is always straight and no further consideration is needed). For one of the directions, the runs have only unit length; we call this the MINOR direction. The other direction, in which the runs can have length longer than 1, is the PRINCIPAL direction. Without loss of generality, we can assume that the C-segment is always in the first octant, that the principal direction is the horizontal one, and that the minor direction is the diagonal one. Such a C-segment can always be represented as a sequence of run lengths along its principal direction. For example, the C-segment in Fig. 9 can be represented as: 1 1 1 1 2 1 2. We call this the RUN LENGTH REPRESENTATION (RLR) of a C-segment or a digital arc which satisfied C1, C2, and C3. Henceforth, we assume that all digital arcs we are dealing with satisfy C1-C3.

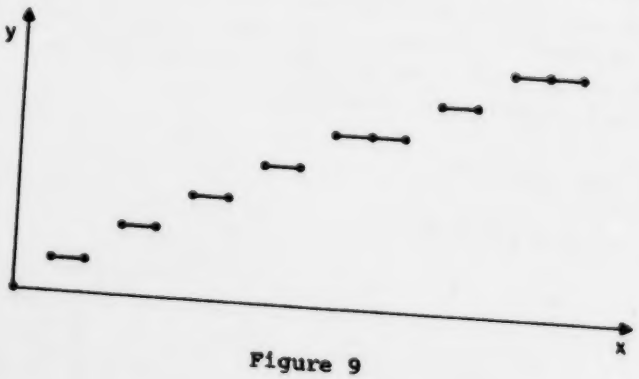


Figure 9

If there is any further critical point in a C-segment, it must be due to the relative relations and distributions of the run lengths in the principal direction. It will be simpler and more convenient to modify the definition of the length of a run along the principal direction to be the NUMBER OF POINTS (all the zero points and the two endpoints) in a run. The new length is equal to the old length plus 1 and is exactly equal to the x-increment of a run. This can be seen in Fig. 9, and the RLR of that arc can be written as 2 2 2 2 3 2 3. It will not add any confusion, because we no longer consider anything in the minor direction.

In considering the possible run lengths in the principal direction of a straight segment, C4 states that there are only two different lengths (if there is only one length, then it is always straight and no further consideration is needed), which are two consecutive integers. C5 considers the run of runs about one of the two lengths and states that it can only occur singly. When considering the run of runs for the other length, Wu in 1980 (15) showed that the possible occurrences of the other length will be two consecutive integers. One of them always occurs singly. The possible occurrences of the other one will again be two consecutive integers, and one of them always occurs singly, and so on. This can be carried on up to many levels until, finally, a single length covers the whole digital arc.

The hierarchical structure of a digital straight line segment had also been shown by Arcelli and Massaratti in 1974 (16). Nevertheless, they restricted their finding to monotonic cases only.

For convenience, we call such a hierarchical structure the consecutive-singly (CS) principle. It can be shown easily by the RLR. At each level, we ignore the run length that occurs only singly (we call it the SEPARATOR). The occurrences of the other one in each block which is separated by the separators are the lengths of the next level. If, at a certain level, both lengths occur more than once, or they differ by more than 1, then this arc is not straight and this process cannot carry on. Otherwise, this process will continue on until there is only one length. For example, the arc in Fig. 9 is not straight, because at the second level two lengths are not consecutive integers.

Example 1: not straight

level 1. 2 2 2 2 3 2 3

level 2. 4 1

Example 2: not straight

level 1. 233233233233233233233233233233233233233233

level 2. 2 2 2 2 3 2 2 2 3 2 2 3

level 3. 4 3 2

Example 3: straight

level 1. 2332332332332332332332332332332332332332333

level 2. 2 2 2 2 3 2 2 2 3 2 2 2 3

level 3. 4 3 3

level 4. 2

The CS principle is excellent in generating a straight digital line segment. However, when it is used to determine the straightness of a digital line segment, there are some problems. Let us consider an example first: for a digital arc CSM, in its RLR:{ 2 3 2 2 3 2 3 2 3 2 3 2 2 3 }, the 3 which occurs only singly is a separator.

level 1. 2 3 2 2 3 2 3 2 3 2 3 2 2 3

[1] level 2. 1 2 1 1 1 2

level 3. 1 3

At level 3, the two lengths are not consecutive but this arc is indeed straight. It seems that the CS principle is more than sufficient, i.e., sufficient but not necessary. However, this is not that case. In ref. 15, it is said that the intervals between two successive occurrences of the one occurring singly must be two consecutive integers. The one that occurs singly at level 1 is 3 and there is no 3 at the left end. The first interval 1 at level 2 should not be there, and we have:

level 1. 2 3 2 2 3 2 3 2 3 2 3 2 2 3

[2] level 2. 2 1 1 1 2

level 3. 3

The CS principle has not been violated. However, we can add a 3 to the left end of the arc and obtain:

```
       level 1.  3 2 3 2 2 3 2 3 2 3 2 3 2 2 3
[3]    level 2.    1   2   1   1   1   2
       level 3.    1           3
```

We still have the same problem. Of course, we can say the same thing at the second level, i.e., there is no 2 at the left end at the level 2. Again we add a 2 there and we have:

```
       level 1.  3 2 2 3 2 3 2 2 3 2 3 2 3 2 3 2 2 3
[4]    level 2.    2   1   2   1   1   1   2
       level 3.        1           3
```

Now this segment is no longer straight, but the problem is still there. If we take off 2 2 3 at the right end we have:

```
       level 1.  3 2 2 3 2 3 2 2 3 2 3 2 3 2 3
[5]    level 2.    2   1   2   1   1   1
       level 3.        1           3
```

Furthermore, we take the 3 off from the left end too and have:

```
       level 1.  2 2 3 2 3 2 2 3 2 3 2 3 2 3
[6]    level 2.  2   1   2   1   1   1
       level 3.      1           3
```

In both cases the segments are not straight; even the separators are missing in both cases. The key point is still the successive occurrences of the other length rather than whether or not there is a separator to end them. Things are more complicated than that. If we move the 3 from the right end to the left end:

```
       level 1.  3 2 2 3 2 3 2 2 3 2 3 2 3 2
[7]    level 2.    2   1   2   1   1   1
       level 3.        1           3
```

the arc is straight again.

In all these examples, the CS principle has been violated in exactly the same way. But some of them are straight and some are not. It seems that if a separator is at the right place, then it will work. Otherwise, it may say a straight segment is non-straight. There is a method which is based on this principle [17]. However, that paper did not show how to deal with these situations, and in all examples in it the segments were long enough that such cases never happened. When dealing with a C-segment or any other limited length segments, we just have to find another way. Of course, we can look into them and try to sort them out if necessary. However, there is a simpler way to overcome them and bypass the chord principle completely.

There is something more fundamental than the CS principle in the structure of a digital straight segment. In those non-straight segments, namely [4], [5], and [6], there are two sections (underlined) in each of them such that each section has the same number of run lengths, but the sum of all run lengths in one section is greater than that of the other section by 2. You cannot find such sections at any length in any of the straight segments, i.e., [1], [2], [3], and [7].

The absence of such uneven sections is the most fundamental structure of a digital straight segment. We will show that in the next section.

VI. NECESSARY AND SUFFICIENT CONDITION FOR THE CHORD PROPERTY

In this section, we are going to give the necessary and sufficient condition for a digital segment to have the chord property. In order to state these clearly, we must give some definitions first.

A C-segment CSM can be represented by its RLR as:

$$CSM = \{ L(1), L(2), \ldots, L(m) \}$$

where the L(i), i = 1, ..., m, are the run lengths in the principle direction; we call them the TERMS of a RLR. For instance, there are m terms in the RLR of CSM.

Definition: A SECTION in a RLR is a segment of consecutive terms of this RLR. The number of terms in a section is its LENGTH, and denoted as LS.

Definition: All sections in a RLR which have the same length are called EQUAL SECTIONS.

Definition: The sum of all terms in a section S1 is called the SUM of section S1 and is denoted as S(S1).

Definition: We say two equal sections are **UNEVEN**, if their sums differ by more than 1. We call any two uneven sections an **UNEVEN PAIR.**

For example, the C-segment in example [6] of Section V has an uneven pair, namely, the equal sections (as underlined in [6]):

$$S1 = \{\ 2\ 2\ 3\ 2\ 3\ 2\ 2\ \} \quad \text{and} \quad S2 = \{\ 3\ 2\ 3\ 2\ 3\ 2\ 3\ \}$$

Their length is LS(S1) = LS(S2) = 7, and their sums are S(S1) = 16 and S(S2) = 18. The length LS of a section is exactly equal to its Y-increment, and its sum S is exactly equal to the X-increment (as in Fig. 10).

Whether there is an uneven pair or not in the RLR of a C-segment is the key factor in determining the straightness of this segment. The following theorem states exactly that.

Theorem T5: A C-segment has the chord property iff there is no uneven pair in its RLR.

Namely, a C-segment is straight iff for any sections in its RLR which have the same length, then their sums cannot differ by more than 1. Before we can prove this theorem, some preliminary remarks and lemmas are necessary.

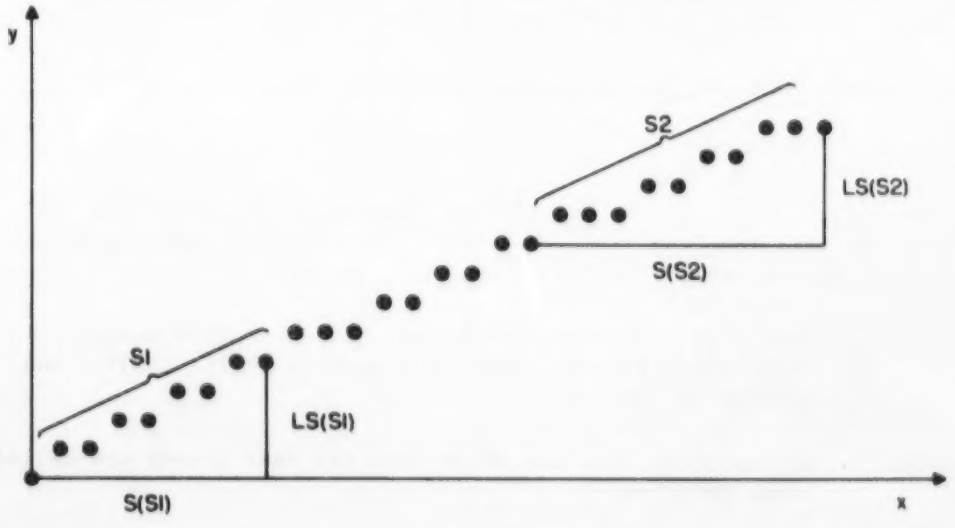


Figure 10

VII. REMARKS AND SOME IMPORTANT LEMMAS

When we consider two equal sections being uneven or not, only those sections with no common terms must be considered. If there is an uneven pair whose two equal sections have some terms in common, then after taking off those common terms they will still be an uneven pair. Consequently, if there are m terms in the RLR of a C-segment, only those sections having lengths up to and less than m/2 need to be considered. More precisely, we have to consider only those sections with length LS such that

$$1 \leq LS \leq M, \text{ where } M = m/2, \text{ if m even, otherwise, } M = (m - 1)/2$$

Let us consider that it is necessary first. If LS = 1, there is only one term in each section. Namely, the run lengths in the principal direction cannot differ by more than 1, or they must be two consecutive integers. This is exactly the C4, and the proof can be found in ref. 12.

Once a C-segment also satisfies C4, there can be only two values for all terms in its RLR, which are two consecutive integers. Let these two integers be n and s = n + 1. If LS = 2, the only equal sections which can be uneven are $\{n, n\}$ and $\{s, s\}$. If this C-segment is straight then one of the n and s can occur only once. This is exactly the C5, and the proof can also be found in ref. 12.

Once a C-segment is tested and no uneven pair can be found for LS = 1 and 2, then it must also satisfy C4 and C5. There are only two values, n and s, in its RLR, and one of them, say s, can occur only singly (if we choose n to occur only singly, the proof is of course analogous). We are going to prove this for LS > 2, and we assume that all segments that we are dealing with satisfy C1-C5.

Lemma 1: Two equal sections are uneven iff the number of occurrences of s in one section is greater than that of the other section by 2 or more.

Proof: Let LS be the length of the equal sections and O(S1) and O(S2) be the numbers of occurrences of s in sections S1 and S2, respectively. Because

$$S(S2) = s*O(S2) + n*[\, LS - O(S2)] = n*LS + O(S2)$$

$$S(S1) = s*O(S1) + n*[\, LS - O(S1)] = n*LS + O(S1)$$

we have

$$S(S2) > S(S1) + 2 \quad \text{iff} \quad O(S2) > O(S1) + 2$$

We call the first and the last terms of a section in a RLR the ENDTERMS, and all the terms between are called the INTERIOR OF A SECTION and denoted as IS.

Lemma 2: If there is an uneven pair, then there always exists an uneven pair such that the section with the greater sum has two longer run lengths as its endterms, the other one having two shorter run lengths as its endterms, i.e., there must be S1 and S2 such that

if $S(S2) > S(S1) + 2$,

then

$S1 = \{n, \ldots IS(S1) \ldots, n\}$, $S2 = \{s, \ldots IS(S2) \ldots, s\}$

Proof: Let $S(S2) > S(S1) + 2$, and n and $s = n + 1$ be the two run lengths. If at one end, S2 has one or more n's beyond the last s, then we can take off those terms from S2, and at the same time take off the same number of terms from S1 (it does not matter if they are s or n). We will still have $S(S2) > S(S1) + 2$. It will be similar if S1 has some s's beyond the last n.

Lemma 3: If there are two equal sections with $S(S2) > S(S1) + 2$ then there must be an uneven pair such that $S(S2) = S(S1) + 2$.

Proof: We can take off as many terms as needed from S2 and the same number of terms from S1, trying to make the $O(S2) = O(S1) + 2$, if S(S2) is not already equal to $S(S1) + 1$. In order to ensure that the S2 has s as endterm we always drop the s and the n-string next to it together. If in some step no s is taken off from S1 than the $S(S2) - S(S1)$ will be smaller by one. We can keep doing this until $S(S2) = S(S1) + 2$. If at some step $S(S2) = S(S1) + 2$, but S1 has an s as end term, then we can have one more step. If in each step, both O(S2) and O(S1) are smaller by one, then at some point S1 must be running out of s. We can drop off all but two s's from S2, and the same number of terms (all are n's now) from S1. The two shortened sections will be the new uneven pair and $S(S2) = S(S1) + 2$.

Henceforth, we assume all uneven pairs that we are dealing with have two equal sections such that their sums differ by exactly 2, i.e.,

$$S(S2) = S(S1) + 2$$

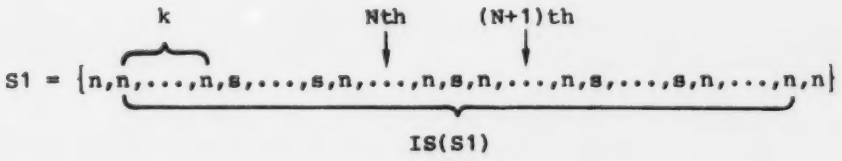
Definition: A MINIMAL PAIR is an uneven pair such that any portions of its two equal sections other than the whole lengths cannot be uneven.

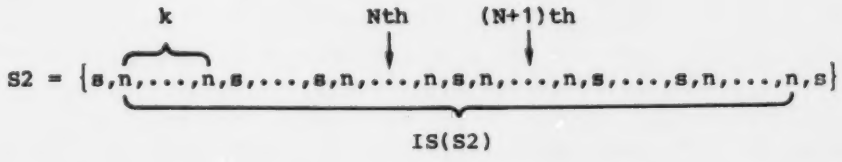
i.e., they will no longer be uneven if we drop off any number of terms from both sections or for any two equal portions in a section.

Lemma 4: The two equal sections, S1 and S2, of a minimal pair have similar structures. More precisely, they have the exact interior but different endterms. If S(S2) = S(S1) + 2 then:

$S1 = \{n,...IS...,n\}$ and $S2 = \{s,...IS...,s\}$

Proof: Let S1 and S2 be:

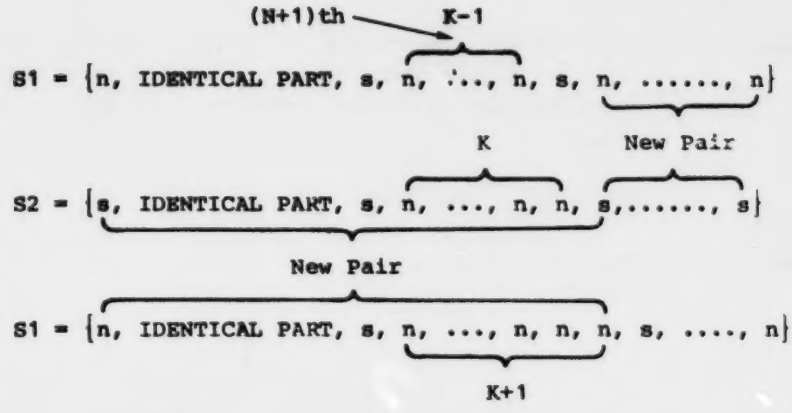




and the first string of n's in IS(S2) has k terms. Then the first n-string in IS(S1) must have length k, too. If the n-string in S1 has k - 1 or less terms, then we can take off the first k + 1 terms from both of them and get a shorter uneven pair (we will take off one s from S2 and one or more s from S1). If the n-string in S1 is k + 1 terms or more, then the first k + 2 terms in both sections will be a pair of shorter uneven sections (for S1 the first k + 2 terms will all be n's and in S2 there will be k n's and two s's in both ends). Contradiction.

We can assume that IS(S1) and IS(S2) are identical up to the Nth n-string. For the (N+1)th n-string, it can have only k, k + 1, or k - 1 terms (if it is k - 2 or less, then the current n-string plus two s's in both sides and the first n-string or part of it will be two shorter uneven sections. For k + 2 or more it is similar). Let the (N+1)th n-string

in S2 have K (k, k + 1, or k - 1) terms. If its counterpart in S1 has K - 1 terms, then we can take off all the terms up to and including the (N + 1)th n-string from S2 and the same plus the s next to this n-string from S1; the rest of them will be a shorter uneven pair.



If its counterpart in S1 has K + 1 terms, then the terms up to and including the (N+1)th n-string in S1 and the same plus the s next to this string in S2 are the new pair of shorter ones. Contradiction; the same argument can be carried out from the other end, too. Therefore, they must be identical all the way.

Lemma 5: Each equal section in a minimal pair is symmetric about its own centre.

Proof: If O(S2) = 2 and O(S1) = 0, then it must be true, because they can have only the forms

$$S1 = \{n, n, \ldots, n, n\} \quad \text{and} \quad S2 = \{s, n, \ldots, n, s\}$$

It is true for O(S2) = 3 and O(S1) = 1, too, because we can only have:

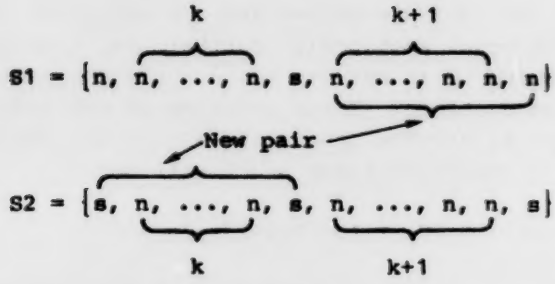
$$S1 = \{n, n, \ldots, n, s, n, \ldots, n, n\}$$

and

$$S2 = \{s, n, \ldots, n, s, n, \ldots, n, s\}$$

and the two n-strings in both sides must have the same number of terms. If they are not the same, say in S2, the

first n-string has k terms and the second one has k + 1 terms. Then by Lemma 4, S1 will have exactly the same IS as S2, and we will have

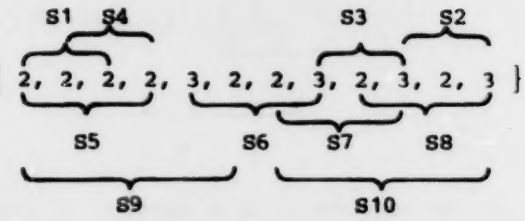


Then the second half of S1 and the first half of S2 (as indicated) will be two new uneven sections. It is similar for k - 1.

For O(S2) > 3. If there is any unsymmetry in S1, say the Nth n-string from the left has length k and the Nth n-string from the right is of k + 1, then the portion of S1 from the right including this n-string and the portion of S2 from the left including the Nth n-string and the s next to it are two shorter uneven sections. Contradiction; and other situations can be handled in a similar way.

Definition: A PRIME PAIR is a minimal pair such that neither of its two equal sections can be moved any closer and still be uneven nor can there be any other minimal pair between them.

For example, in the segment CSM:



(S1, S2), (S1, S3), (S4, S2) and (S4, S3) are all minimal pairs, but only (S4, S3) is prime. (S5, S7) and (S5, S8) are uneven but not minimal, while (S5, S6) is minimal and also prime. (S9, S10) is uneven but nothing else.

It is obvious that if there is an uneven pair either it is a prime pair, or we can always find a prime pair. Actually, the uneven pairs which

will be found by the algorithm that we will give in the next section are prime pairs. Thus, we need only to deal with the prime pairs, henceforth.

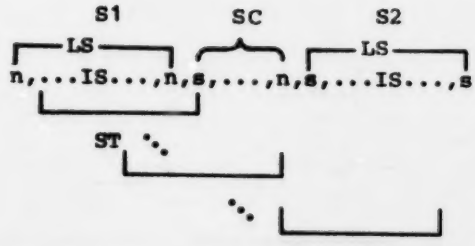
Lemma 6: The number of terms between two uneven sections of a prime pair (S1, S2) are either zero or multiples of LS (the length of the equal sections). Furthermore, each of those sections of length LS in between has exactly the same IS as the uneven sections. Their endterms at the ends near the S1 are always s, and the others are always n. Consequently, the sums of those sections S(SC) will be:

$$S(S2) = S(SC) + 1 = S(S1) + 2$$

Proof: If there is any term between S1 and S2, it can be neither s nor n alone (otherwise, S1 or S2 can move closer). So it must be at least an s in the S1 side and an n in the S2 side. There cannot be just one s and one n, otherwise {s, n, s} and any {n, n, n} in S1 or S2 will be two shorter and closer uneven sections. Unless there is no {n, n, n} in S1 and S2, then S1 and S2 can only be:

$$\{n, n\},\{s,s\} \quad \text{and} \quad \{n,n,s,n,\ldots,s,n,s,n,n\},\ \{s,n,s,\ldots,n,s,n,s\}$$

If it is the first case, then we are finished, i.e., the common IS has zero term and the centre section has s and n as endterms and SC = {s, n}. In the second case, S2 can move one step closer to S1. A contradiction, so we can assume that there are some more terms between and we will have

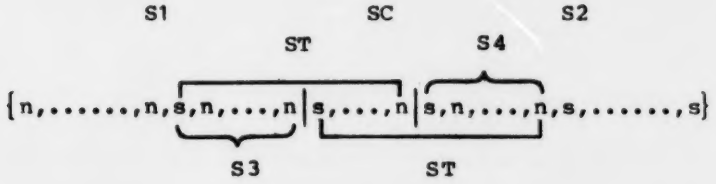


where ST can be considered as a section in travel. In each step, it moves one term away from S1 and towards S2 by taking off its left endterm and picking up the term that is next to its right endterm. The first ST is obtained by taking off the first n from S1 and picking up the first s

from SC. The last ST, after it takes off its left endterm n and picks up the last term s of S2, then becomes S2. Those are the only occasions that a ST can drop off an n and pick up an s. In all other steps, the term being taken off and the term that is going to be picked up must be identical. Otherwise, S1 and S2 cannot be prime. Because, for each ST it must have

$$LS(ST) = LS \quad \text{and} \quad S(ST) = S(S1) + 1$$

If for some ST $S(ST) > S(S1) + 1$ (or $S(ST) < S(S1) + 1$), then the ST and S1 (or S2) will be a new uneven pair; and, if in any step, ST drops an n (or s) and picks an s (or n), then its sum will increase 1 (or decrease 1). Then SC must be identical with S1 term by term, except the first term, and its length will be at least equal to LS. If LS(SC) is less than LS, then there are some ST's which will cover terms from both S1 and S2, and there will be two identical portions in S1 and S2 as indicated in the following:



S3 is identical to S4, so (S1 - S3) and (S2 - S4) will be a new uneven pair. This is a contradiction and SC should be at least the same length as S1. The same argument can be applied to the portion of SC which is longer than LS and also can be applied from the S2 side. Therefore, SC must be either zero terms or have length equal to the multiples of LS and each LS section must have the same structure.

VIII. THE PROOF OF THE MAIN THEOREM T5

Now we are going to give the proof for the necessary part of T5, i.e., if there is an uneven pair in the RLR of a digital segment, then there must be two digital points in this segment such that in the line segment connecting these two digital points, there must be at least one point whose distance to the nearest digital point is equal to or greater than 1. We only have to show that this is true for uneven pairs of length >2.

We must make one more remark concerning the first or the last term of a digital segment. If the first or the last term of a segment is a shorter run (as n in our example), such a run can be counted into the RLR only if the endpoint is not located in it. In Fig. 11a, the first run should be counted, but not in (b). Consequently, the segment in (b) is straight but not the one in (a).

More generally, if the first term of a digital segment is shorter than the next term or if it is equal to the next term or next few terms but all of them are shorter than the one following, then the first term can only be counted if the endpoint is not located on it. The situation is similar for the last term.

Now we give the proof for the necessary part of T5 as follows: If there is an uneven pair, say S1 and S2, in the RLR of a digital segment, we can assume that it is a prime pair, and S1 is in the left of S2. If we join the last point A (the right endpoint) of S2 and the point F just before the first point of S1 as indicated in Fig. 12, we can also assume, by Lemma 6, that there are M equal sections of length LS between S1 and S2. Then the point C in $\overline{AF}$ whose X-coordinate X(C) is

$$S(S1) + M*[S(S1) + 1] + 1$$

will have a Y-coordinate Y(C) which will satisfy

$$Y(C)/[(M + 2)*LS] = X(C)/\{S(S1) + S(S2) + M*[S(S1) + 1]\}$$

We have

$$Y(C) = (M + 1)*LS$$

and

$$X(C) = [M + 1]*[S(S1) + 1]$$

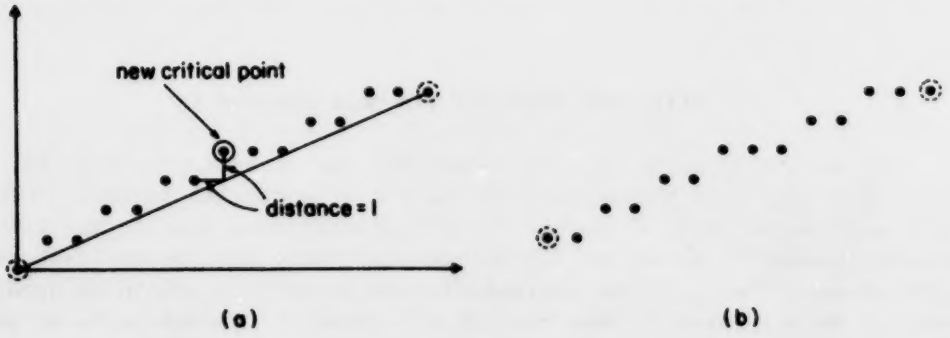


Figure 11

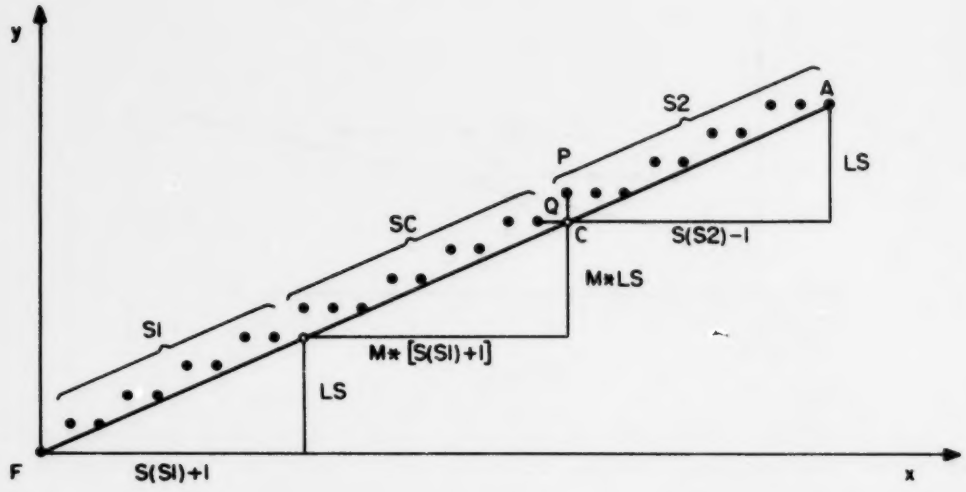


Figure 12

The first point P of S2 at the left has the same X-coordinate C but its Y-coordinate is just one more. The point Q which is just before P has the same Y-coordinate as C, but its X-coordinate is one less than X(C). At point C the chord property is violated. Namely, if there is an uneven pair, we can always find a point such that the chord property is violated, so the condition is necessary.

The condition is also sufficient. We will show the equivalent: if a segment does not have the chord property then there must be an uneven pair in its RLR.

If a digital segment does not have the chord property, then there must be two digital points in it such that, in the line segment joining those two points, there must be at least one point whose distance to the nearest digital points is equal to or greater than 1. The distance from a point P = (x, y) in the line segment joining the two digital points to a digital point D = (h, k) is equal to:

$$\max \{ |x-h|, |y-k| \} \quad \text{or} \quad \max \{ X(\overline{PD}), Y(\overline{PD}) \}$$

Let point A and point F be two such digital points in the digital segment, and there is a point P in $\overline{AF}$ such that there is no digital point whose distance to P is less than 1. We have to consider only two cases, namely, for P and a digital point D such that X(P) = X(D), $Y(\overline{PD}) = 1$ and $Y(\overline{PD}) > 1$, where digital point D is a left endpoint of a run in the digital segment (see Fig. 13).

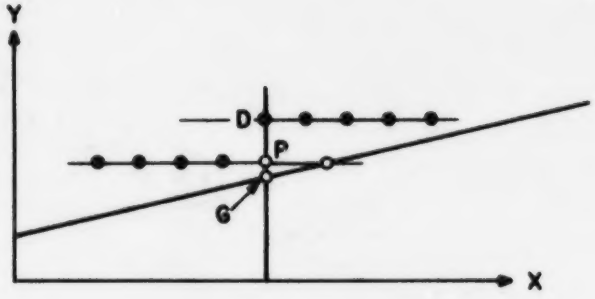


Figure 13

If P is in the right side of D, i.e., X(P) > X(D), and has a distance equal to or greater than 1 along the Y-direction with any digital point, then there must be a point G in $\overline{AF}$ such that X(G) = X(D) and Y($\overline{GD}$) > 1. If P is in the left side of D and violates the chord property, then we will end up in the same situation but with the left endpoint of the next run.

First, we will deal with the case that X(P) = X(D) and Y($\overline{PD}$) = 1. We can assume that A and F are the closest digital points to do so. Otherwise, if there is another pair between with the same property, then we can apply the same argument to the new pair. Since both X(P) and Y(P) are integers, there must be a point E in $\overline{AF}$ such that $\overline{AP}$ = $\overline{EF}$ and X(E) and Y(E) are also integers (see Fig. 14). Furthermore, E cannot be a point in the segment; otherwise, A and F cannot be the closest pair to do so; and E must be just under the left endpoint of a run. Otherwise, we will have a point in AF which has distances greater than 1 to all digital points in this segment; this will be considered later.

From Fig. 14, since Y($\overline{AP}$) = Y($\overline{EF}$), the number of runs in S2, which consists of the digital points whose Y-coordinates are greater than Y(P) and less than or equal to Y(A), and the number of runs in S1, which consists of all digital points whose Y-coordinates are greater than Y(F) and less than or equal to Y(E), must be equal and LS(S2) = Y($\overline{AP}$) = Y($\overline{EF}$) = LS(S1). However, the X-increments of S1 and S2 will be:

$$S(S1) = X(\overline{EF}) - 1 \quad \text{and} \quad S(S2) = X(\overline{AP}) + 1$$

then

$$S(S1) + 2 = S(S2)$$

S1 and S2 are two uneven sections.

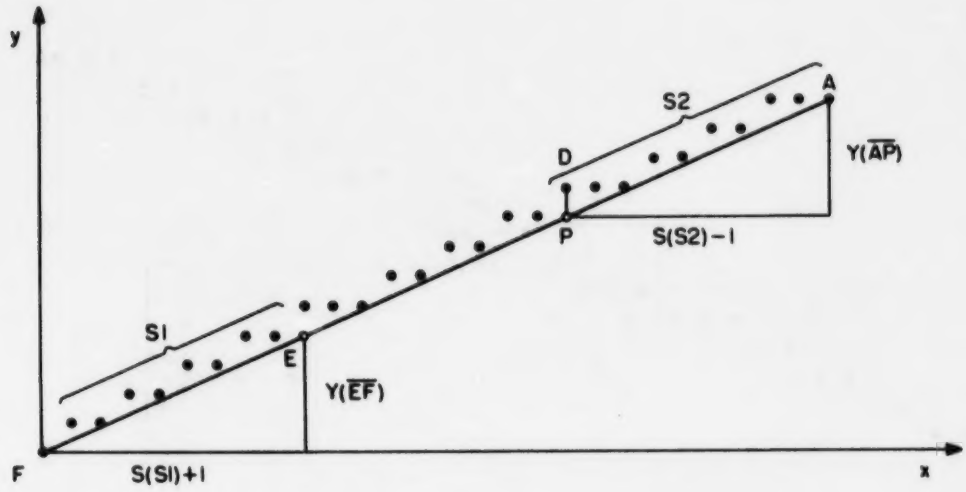


Figure 14

Now we will prove the second case that X(P) = X(D) but $Y(\overline{PD}) > 1$. There must be a point C in $\overline{AF}$ such that X(C) > X(P) and Y(C) is an integer just above Y(P) (see Fig. 15).

Let LS = Y(A) - Y(C). There must a point E in AF such that Y(E) - Y(F) = LS and Y(E) must be an integer. If E is exactly on H or at the right of H, where H is a point whose X-coordinate, X(H), is equal to that of the left endpoint of the run just next to E at the right side and Y(H) = Y(E), then we have X(E) = X(H) or X(E) > X(H) and

$$S(S1) + 1 \leq X(\overline{EF}) = X(\overline{AC}) \leq S(S2) - 1$$

then

$$S(S1) + 2 \leq S(S2)$$

where S1 is the section which consists of all runs between E and F (i.e., all digital points with Y-coordinates greater than Y(F) but less than or equal to Y(E)), and S2 is the section which consists of all runs between A and C.

If X(E) < X(H), i.e., E is on the left of H, then $X(\overline{IE}) = X(\overline{CJ})$. Because GE = CB and their integer portions $\overline{GI} = \overline{JB}$, too, we have

$$S(S1) = X(\overline{FE}) - X(\overline{IE}) = X(\overline{CB}) - (\overline{CJ}) = S(S2) - 2$$

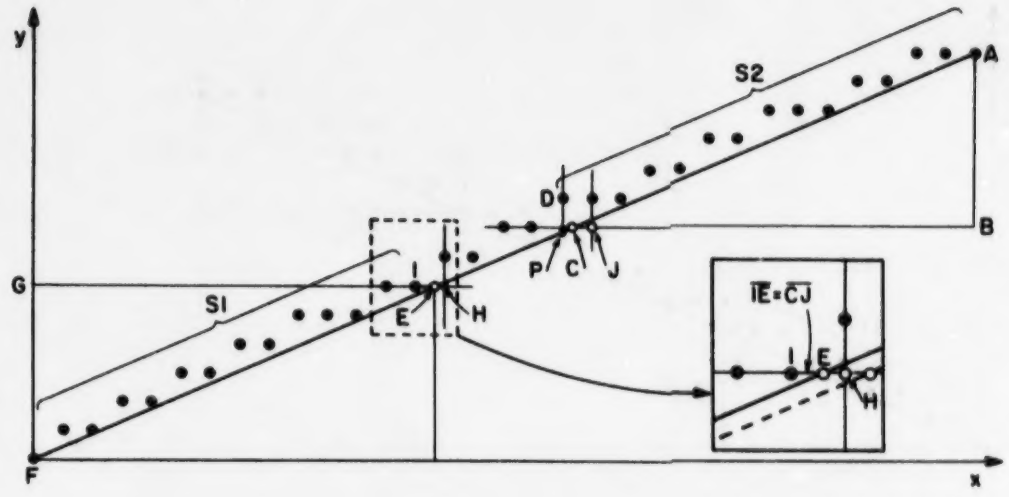


Figure 15

then

$$S(S1) + 2 = S(S2)$$

and S1 and S2 are two uneven sections, and the proof is complete.

After we prove Theorem T5, we simply reduce the multiple level expression of the hierarchical structure of the digital straight segments to the ground level. The uniformity of the chain codes of such segments can then be described as: for all sections having the same number of runs, the sum of all their run lengths cannot differ by more than 1. Namely, for any two sections in such a segment with the same Y-increments, then their X-increments cannot differ by more than 1. The algorithm which is based on this theorem will be used to find further critical points in a C-segment. It is intended to be simple to use and easy to understand.

IX. POSSIBLE CRITICAL POINTS IN A C-SEGMENT

The checking of a C-segment for T5 can be done after all C-segments are determined or immediately after a new critical point is found, i.e., a C-segment is determined. The only information we must know is the RLR of the segment.

Once we have the RLR of a C-segment, we can quickly eliminate some

obvious cases from which no more new critical point can be found and which must satisfy T5:

1. There is no zero point between two endpoints.

2. All run lengths (terms) in the RLR are the same, or there is only one run.

For all other C-segments we must check all the equal sections having length up to M, where M = m/2, m is even, and M = (m - 1)/2 otherwise, and m is the length of this segment.

The procedure consists of two steps: let

$$CSM = \{L(1), L(2), \ldots, L(m)\}$$

be the RLR of the segment.

1. To calculate the sums of all equal sections:

For all sections of length 1:

$$S(1, i) = L(i), \quad i = 1, \ldots, m$$

For all sections of length K > 1, i.e., K = 2, 3, ..., M:

$$S(K, 1) = L(1) + L(2) + \ldots + L(K)$$

$$S(K, 2) = S(K, 1) - L(1) + L(K + 1)$$

$$S(K, 3) = S(K, 2) - L(2) + L(K + 2)$$

$$\vdots$$

$$S(K, m - K + 1) = S(K, m - K) - L(m - K) + L(m)$$

For example, in Fig. 11a, RLR = {2 2 2 3 2 3} and m = 6, M = 3.

$$S(1, i) = 2, 2, 2, 3, 2, 3$$

$$S(2, i) = \quad 4, 4, 5, 5, 5$$

$$S(3, i) = \qquad 6, 7, 7, 8$$

We can also calculate S(K, i) from S(K - 1, i) without subtracting:

CSM = { L(1), L(2), L(3), L(4),.....,L(M),.....,L(m) }

K = 1 S(1,1),S(1,2),S(1,3),S(1,4),...,S(1,M),.....,S(1,m)

K = 2 S(2,1),S(2,2),S(2,3),...,S(2,M-1),...,S(2,m-1)

K = 3 S(3,1),S(3,2),...,S(3,M-2),...,S(3,m-2)

K = 4 S(4,1),...,S(4,M-3),...,S(4,m-3)

. . . .

. . . .

. . . .

K = M S(M,1),.....,S(M,m-M+1)

where S(1, i) = L(i) and each S(K, i) = S(K - 1, i) + L(i + K - 1).

2. To check the difference between any two S(K, i) with the same K. This can be done in two ways: check all possible K's for an i, then go to the next i; or for each K, check all i's then go to the next K. However, it is more desirable to check every K separately, because most often the condition will be violated at a lower level of K, i.e., when the equal sections have only 1 or 2 terms. Once a new critical point is found, we divide a C-segment into two and check them separately.

Let P(K, 1) and P(K, 2) be two values that appear first in S(K, i) and P(K, 1) < P(K, 2). If we are also interested in knowing the pair of uneven sections as well as finding the critical point, then we have to use two more parameters A(K, 1) and A(K, 2), which are used to indicate the addresses of the last occurrences of P(K, 1) and P(K, 2), respectively, i.e., the last i for which S(K, i) = P(K, 1) or P(K, 2). If we check each K separately, then we can use P(1), P(2) and A(1), A(2) rather than the K.

To use this process, let

P(K, 1) = 0, P(K, 2) = 0, A(K, 1) = 0, A(K, 2) = 0

In the following steps, we write out only the values that have been changed. If the values have not been mentioned, that means they are the same as in the previous step. The section whose sum is S(K, i) is denoted as SM(K, i).

For i = 1, let P(K, 1) = S(K, 1), and A(K, 1) = 1.

For i = 2, if S(K, 2) = P(K, 1), then A(K, 1) = 2.

if S(K, 2) = P(K, 1) + 1, then P(K, 2) = S(K, 2),
A(K, 2) = 2.

if S(K, 2) = P(K, 1) - 1, then P(K, 2) = S(K, 1),
A(K, 2) = 1,
and P(K, 1) = S(K, 2),
A(K, 1) = 2.

if S(K, 2) > P(K, 1) + 1,

or S(K, 2) < P(K, 1) - 1, then a critical point is found. The first point of the section SM(K, 2) is chosen to be critical. SM(K, 1) and SM(K, 2) are two uneven sections.

This step should be carried on until P(K, 2) is filled by a value other than zero, unless it ends up in a new critical point. If we run out of S(K, i) before that happens then no critical point can be found for this K; proceed to the next K.

We assume that when i = G, P(K, 1) and P(K, 2) are filled with values other than zero, obviously P(K, 1) + 1 = P(K, 2). The previous step is actually for all i = 2, ..., G.

For i = (G + 1),(G + 2), ..., (m - K + 1)

if S(K, i) = P(K, 1), then A(K, 1) = i.

if S(K, i) = P(K, 2), then A(K, 2) = i.

if S(K, i) < P(K, 1), then a critical point is found. The first point of the section SM(K, i) is chosen to be a critical point. SM(K, i) and SM(K, j) are two uneven sections, where j = A(K, 2).

if S(K, i) > P(K, 2), then a critical point is found. The first point of the section SM(K, i) is chosen to be a critical point. SM(K, i) and SM(K, j) are two uneven sections, where j = A(K, 1).

For example, the segment in Fig. 11a has a RLR = { 2, 2, 2, 3, 2, 3}.

For K = 1, no critical point is found, and the parameters end up as:

P(1, 1) = 2, P(1, 2) = 3, A(K, 1) = 5, A(K, 2) = 6.

For K = 2, no critical point is found, and the parameters end up as:

P(2, 1) = 4, P(2, 2) = 5, A(K, 1) = 2, A(K, 2) = 5.

For K = 3, a critical point is found when:

P(3, 1) = 6, P(3, 2) = 7, A(K, 1) = 1, A(K, 2) = 3,

but S(3, 4) = 8 and we have: S(3, 4) > P(3, 2).

The first point of the fourth equal section of length 3, i.e., SM(3,4) is chosen as indicated in Fig. 10a. SM(3, 1) and SM(3, 4) are two uneven sections.

It is not necessary to know the A(K, 1) and A(K, 2) to determine the critical point. If one is only interested in finding the critical point, then the A(K, 1) and A(K, 2) can be completely ignored. The pair of uneven sections determined by this algorithm is a prime pair, except perhaps when K = 1.

Once a critical point is located, the C-segment is divided into two. For the first subsegment, the checking can be carried on from the next K in line. However, we only have to do this up to m'/2 or (m' - 1)/2, where m' is the number of terms preceding this new critical point. The second subsequent should be treated as a new one, but there are at most m" = m - m' terms in it. In Fig. 4, the point indicated by * is a critical point found by this procedure.

After this procedure, the final C-segments are all guaranteed to have the chord property. A digital arc is segmented into a set of connected straight segments. How to use them is up to the user.

X. SOME PRACTICAL CONSIDERATIONS

The definition of digital arcs in the beginning of Section II is widely used and perfectly correct theoretically. In practice, however, it did cause some problems. There are two obvious ones:

1. It cannot represent a right angle accurately, if the two sides of a right angle are along the horizontal and vertical directions. The corner point of such a right angle is always missing, as in Fig. 16a. Otherwise, the points a and b will have three other points in their 8-neighbourhoods. When b and c are chosen as critical points, the lines that connect them will not meet at a right angle, although they are only off by less than one unit length to the original pixels.

2. The point b in Fig. 16b has a curvature code 2 or -2. It must be a critical point. However, in many cases we want it to be seen as a noise point, especially when the digital segments in both sides of b are long enough.

This can be overcome easily by adding a corner point, in case 1, or replacing the point by one that is in line with the others, in case 2. However, we must make sure that we do it only when it is desirable and will not cause any extra problem.

In both cases, we only make a change when the digital segments in both sides of such a point have at least three points. If one or both digital segments have less than three points, we are not quite sure it will be desirable to make a change, e.g., in case 1, if one side is of length 2, as in Fig. 17.

This is a case of SPCP and has been considered in Theorem T3. The point a is an ideal critical point and there is no need to make any modification.

If both sides have at least three points, then the segment we are concerned with is isolated by zero points from both sides and can be identified easily. In case 1, when there is a NZS of 1, 1 or -1, -1 pair which is isolated by zero points, we can add a point between, as in Fig. 18a. The curvature code of the newly added point will be equal to the sum of that of the two points in both sides, and these two non-zero points will become zero points in the modified digital arc.

Figure 16

Figure 17

Because the problem only occurs when the two sides of a right angle are in the horizontal and vertical directions, we must make sure that the sides are not in the diagonal directions. Otherwise, there will be no room for a new point, as in Fig. 18b. This can be done by checking the chain codes of the two non-zero points. The first one should be even, and the second one should be odd (from even to odd). When both sides are along the diagonal directions, the first turn must be from diagonal to horizontal or vertical, thus the chain code of the first point must be odd (i.e., 1, 3, 5, 7), and the second one must be even (from odd to even).

In case 2, we simply replace the point in question by a point in line with the others, if there is a NZS of the form 1, -2, 1 or -1, 2, -1, as in Fig. 19. After the modification, the sum of the curvature codes of these three points remains zero. As in case 1, the sums of curvature codes before and after the change are the same; this modification will not cause any extra problem in the global situation.

Since the original digital arc satisfies the definition we gave in Section II, case 2 cannot happen in diagonal directions and no further checking is needed.

All these modifications should be done after the curvature codes are known, but before the process goes any further.

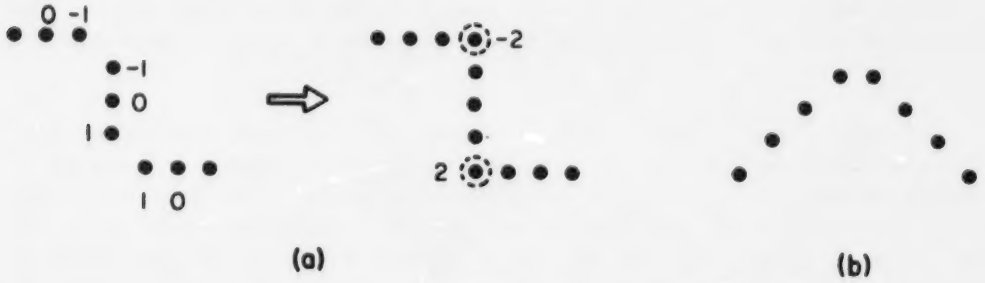


Figure 18

Figure 19

XI. CONCLUSION

The necessary and sufficient condition for a digital arc to have the chord property as well as the method to check them are given. They provided a much simpler alternative to check the straightness of the digital arc, compared to testing the definition of the chord property directly. The first part of the algorithm, namely to check the C1-C3, is based on the local pixel logic. By using the curvature code, we can avoid all complicated angle and distance calculations, as well as any mathematical tool except simple arithmetic. The second part is based on Theorem T5. Theorem T5 gave the most comprehensive and precise expression about the third criterion of digital straight line of Freeman. It reduced the multiple level expression of the hierarchical structure to one level and gave the precise description about the term "as uniformly as possible". Consequently, the algorithm that is based on it is simple to understand and fast to process. It will require a computer to calculate some simple arithmetic but very few decisions are necessary. The final result is a set of connected digital straight line segments and it can used as the base for other processes.

REFERENCES

1. R. Bellman. On the approximation of curves by line segments using dynamic programming. Commun. ACM, 4: 284; 1961.

2. B. Gluss. A line segment curve-fitting algorithm related to optimal encoding of information. Inf. Control, 5: 261-267; 1962.

3. T. Pavlidis. Waveform segmentation through functional approximation, IEEE Trans. Comput. C-22: 689-697; 1973.

4. T. Pavlidis. Segmentation of plane curves. IEEE Trans. Comput. C-23: 860-870; 1974.

5. U. Ramer. An iterative procedure for the polygonal approximation of plane curves. Comput. Graphics Image Process. 1: 244-256; 1972.

6. J. Sklansky, R.L. Chazin, and B.J. Hansen. Minimum perimeter polygons of digitized silhouettes. IEEE Trans. Comput. C-21: 260-268; 1972.

7. P.V. Sankar and C.U. Sharma. A parallel procedure for detection of dominant points on a digital curve. Comput. Graphics Image Process. 7: 403-412; 1978.

8. H. Freeman and L.S. Davis. A corner-finding algorithm for chain coded curves. IEEE Trans. Comput. C-26: 297-303; 1977.

9. A. Rosenfeld and E. Johnston. Angle detection on digital curves. IEEE Trans. Comput. C-22: 875-878; 1977.

10. J. Sklansky and V. Gonzalez. Fast polygonal approximation of digitized curves. Pattern Recognition, 12: 327-331; 1980.

11. H.C. Lee and K.S. Fu. Using the FFT to determine digital straight line chain codes. Comput. Graphics Image Process. 18: 359-368; 1982.

12. A. Rosenfeld. Digital straight line segments. IEEE Trans. Comput. C-23: 1264-1269; 1974.

13. S.H.Y. Hung and T. Kasvand. Critical points on a perfectly 8- or 6-connected thin binary line. J. Pattern Recognition, 16(3): 297-306; 1983.

14. H. Freeman. Boundary encoding and processing. Picture Processing and Psychopictorics. Edited by B.S. Lipkin and A. Rosenfeld, Academic Press, New York, NY. 1970. pp. 241-266.

15. Li-De Wu. On the Freeman's conjecture about the chain code of a line. Proc. 5th International Conf. on Pattern Recognition. 1980. pp. 32-34.

16. C. Arcelli and A. Massarotti. Regular arcs in digital contours. Compt. Graphics Image Process. 4: 339-360; 1975.

17. S. Shlien. Segmentation of digital curves using linguistic technique. Comput. Graphics Image Process. 22: 277-286; 1983.